AFRL-VS-TR-2001-1577

CBSD: THE CELESTIAL BACKGROUND SCENE DESCRIPTOR VERSION 4.0 – INSTALLATION AND USER'S MANUAL

Paul V. Noah
Meg A. Noah

Mission Research Corporation
589 West Hollis Street, Suite 201
Nashua, NH 03062-1323

February 2001

Scientific Report No. 7

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AIR FORCE RESEARCH LABORATORY
Space Vehicles Directorate
29 Randolph Rd
AIR FORCE MATERIEL COMMAND
Hanscom AFB, MA 01731-3010

This technical report has been reviewed and is approved for publication.

Contract Manager

Branch Chief

This report has been reviewed by the ESC Public Affairs Office (ESC/PAM) and is releasable to the National Technical Information Service (NTIS).

Qualified requestors may obtain additional copies from the Defense Technical Information Center (DTIC). * (All others should apply to the NTIS.)

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by you, please notify AFRL/VSIM, 29 Randolph Road, Hanscom AFB, MA 01731-3010. This will help us to maintain a current mailing list.

DO NOT RETURN COPIES OF THIS REPORT unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB NO. 0704.0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE February 2001	3. REPORT TYPE AND DATES COVERED Scientific, Interim 1 Apr 98 - 31 Mar 99

4. TITLE AND SUBTITLE CBSD: The Celestial Background Scene Descriptor Version 4.0 - Installation and User's Manual	5. FUNDING NUMBERS Contract: F19628-93-C-0028 PE: 63215C PR: S321 TA: GG WU: AA
6. AUTHOR(s) Paul Noah Meg Noah Mission Research Corp. – Nashua, NH 03062	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Mission Research Corporation P.O. Drawer 719 Santa Barbara, CA 93102-0719	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory 29 Randolph Road Hanscom AFB, MA 01731 Contract Manager: Dr. Steven Price	10. SPONSORING/MONITORING AGENCY REPORT NUMBER AFRL/VS-TR-2001-1577

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited.	

ABSTRACT *(Maximum 200 words)*

This document is a User's Manual for two of the Celestial Background Scene Descriptor (CBSD) models; CBZODY6; the zodiacal light model; and CBSKY4, the point source stellar component. The codes provide flux and postional information for stars and zodiacal dust covering the 0.2 to 30μm range. The CBZODY6 from 0.2 to 30μm and the CBSKY4 from 2 to 30μm can use external filter response functions of any size. Below 2μm, the CBSKY4 model is limited to discrete bands: Johnson (BVJHK); the COBE/DIRBE 2.4μm band (Band 2); IRAS 12 and 25um (Bands 1 and 2); TD1A 1565 Å band; FAUST 1660 Å band; and the Apollo 16 "S201" 1400 Å band. The CBZODY6 model gives the zodiacal contribution for any near Earth viewing geometry. The CBSKY4 model gives position of the 2552 brightest stars from the IRAS and MSX catalogs. Over 9000 stars in the visible are taken from the Tale Bright Star Catalog. Additionally, the CBSKY4 code uses a statistical model to generate realistic scenes to any magnitude.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Celestial, Infrared, Visible, Modeling, Stars, Zodiacal	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	None

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18
298-102

Table of Contents

1. Overview

1.1 Introduction.

MRC/Nashua is developing a set of codes that generate two-dimensional radiance maps for any region of the galactic background for the Air Force Research Laboratory, Battlespace Environment Division at Hanscom Air Force Base, MA. Known as the Celestial Background Scene Descriptor (CBSD), these models are stand alone FORTRAN codes designed to provide images, line-of-sight spectra, or line-of-sight integrated radiances. Designed for evaluation of bandpass sensor systems, these models currently operate over the visible and infrared regions of the spectrum (defined here as the 0.2μm to 35μm region). Two models are presently implemented in CBSD Version 4:

1. A galactic point source model (CBSKY4), and
2. A model of the interplanetary dust cloud (CBZODY6).

Additional models to be implemented include:

1. The distribution of known asteroids.
2. A statistical distribution of unobserved asteroids (expected numbers),
3. Galactic extended sources (planetary nebula and reflection nebula),
4. Extended extragalactic sources (galaxies and quasars), and
5. A solar system module supporting the Sun, Moon, planets, and planetary satellites.

The complete suite of models provides a realistic view of the celestial background.

There is an additional module referred to as CBSD4 which is the control program allowing all models to be called from a single input file. This assures that the different scenes are spatially and temporally consistent. The module also sums the components into single self-consistent scene and optionally convolves the image with a gaussian point spread function.

1.2 Galactic Point Sources.

The emissions from stellar point sources represent the most important source of clutter in the celestial background. Most point source models use observational data, principally the IRAS (the Infrared Astronomical Satellite) catalog of point sources. Even though the IRAS catalog contains 246,000 objects, it has limited spectral coverage (the entire sky was sampled at only 12 and 25μm within the 0.2 - 35μm region). IRAS had a large pixel size, limiting the flux of individual sources and undersampling the star count in the densest parts of the galactic plane.

The CBSD galactic point source model, CBSKY4, generates two-dimensional radiance maps of stellar point source fluxes. To provide realistic star counts over the entire sky, the Air Force Research Laboratory, Space Vehicles Directorate has produced the SKY4 model. The SKY4 model provides a single line-of-sight statistical distribution of the number of point sources per unit angular area per magnitude interval for each of 87 spectral classes (Wainscoat et al. 1992, Cohen 1993, 1994a, 1994b). CBSKY4 uses the SKY4 model for populating local regions with stars. In the CBSD implementation, the brightest objects in the statistical model have been removed and replaced with the 2,503 objects observed by IRAS and 49 stars from MSX. In the visible, the Yale Bright Star Catalog is used with over 9,000 objects. Also, the CBSD point source model includes an estimate of the interstellar extinction, an important consideration for short wavelength studies.

1.3 Solar System Dust Cloud.

The solar system dust cloud, also known as the zodiacal dust, is seen as thermal emission over all the sky and dominates the diffuse IR emission over most of the 4 - 35μm waveband. Below 4μm, scattered sunlight becomes an important contributor to the background. The Zodiacal dust is made up of two clearly observable structures, the broad emission and the band structure. The broad emission is that component seen over the entire sky, being strongest close to the plane of the ecliptic (Sykes 1988, 1990). It is theorized that a dust cloud, composed of debris from asteroid collisions and comets, encircles the Sun (Dermott et al., 1984). The Earth is located inside this cloud, which causes the emission to be visible over the entire sky.

The band structure, discovered by IRAS in 1983, consists of several pairs of faint bands of emission close to the plane of the ecliptic associated with the remnants of the breakup of individual asteroid families (Sykes et al., 1989, Dermott et al. 1985, 1986). The CBSD Zodiacal light model, CBZODY, models both the broad and band contributions, including thermal emission and scattered light, and is capable of producing spectra, line-of-sight radiance, and images. The model CBZODY also generates single line-of-sight radiances for specified coordinates.

1.4 Using this manual

This manual is divided into five sections. Section 2 discusses the installation of the various CBSD components, Sections 3 and 4 provide the details of running the individual codes. Section 5 gives some troubleshooting tips.

1.5 FITS

The CBSD codes are standard ANSI FORTRAN and have no graphical interfaces or display capabilities. Image output is provided in the Flexible Image Transport System (FITS) format. This allows the maximum compatibility with various platforms.

http://fits.gsfc.nasa.gov/

The FITS support office at NASA.

http://fits.gsfc.nasa.gov/fits_intro.html

A brief introduction to FITS.

http://www.cv.nrao.edu/fits/FITS.html

This is an archive for the Flexible Image Transport System [FITS], the standard data interchange and archival format of the worldwide astronomy community.

http://asia.yahoo.com/science/astronomy/fits__flexible_image_transport_system_/

This site has links to other FITS standards sites.

http://www.scispy.com/Standards/FITS.html

Description of FITS.

http://www.nso.noao.edu/diglib/formats.html

The National Solar Observatory maintains FITS viewer libraries. For users with a Unix or VMS system on any platform including PCs, the XV package can simply display FITS format files. You can also use the IRAF package for reading, displaying, and analyzing FITS images.

1.6 Input File Conventions

All inputs for the codes are derived from sets of ASCII input files such as CBSKY4.INP or CBZODY6.INP. These input files are broken into logical sections of related items. The section heading identifies the section. Under the section heading is a list of required keywords and optional keywords.

The input files for the CBSD codes follow the standard Microsoft Windows© ini format. That is, parameters are grouped by category. Each category is assigned a section name that is set off by brackets. Each line under that category consists of a line where

Keyword = *value*

1. Where ***keyword*** is a predefined parameter name and *value* is the user defined parameter value. For example:

   [path]

   ; This is a comment line.

   Path = *c:\cbsd*

2. Furthermore, comments are indicated by semicolons (;) and may appear anywhere in the input file. However, once a comment symbol appears on a line, no further parsing of that line is performed.

The guidelines for identifying the headings, parameters and input values, followed in all the input files, are given below:

Keyword	Usage
[image]	- Bold, bracketed entries indicate section headings.
pixel_size	- Bold, italic entries are input parameters.
Yes	- Italic entries are responses for each parameter.
"Path"	- Italic, quoted entries are text entries and are usually user specified path or filenames.
Integer	- Italic, underscored entries are numerical input in the indicated format.

2 Installation

2.1 Preparation

The CBSD Version 4.0 distribution comes on a single ISO-9660 compatible CD-ROM containing 28 Megabytes of data files, source code, and executables. There are four components on the CD-ROM:

1. All source code and ASCII data files. (PC)
2. The PC executables files
3. The PC unformatted (binary) data files
4. Unix compressed, tar file of all source code and ASCII data.

2.2 Windows 98, Windows NT, Windows 2000 Installation

For the PC version, a complete installation of source code, executable, ASCII data, and unformatted data, items 1, 2 and 3, only requires 30 Megabytes of disk space, however, additional disk space is needed to build the executables and to store the voluminous output these codes can produce.

A minimum PC installation needs 9 Megabytes for the executables. The CBSD PC executables can run with the unformatted data files resident on the CD-ROM. Additional free space is required for output data and images. The source code distribution requires 12 Megabytes of free memory for execution.

We recommend two parallel directory structures, one for the source code and ASCII version of data files, and one for the executables. This minimizes the number of files to track and allows the deletion of the entire source directory tree when the executables and data files are built. The recommended CBSD directory structure is:

```
\cbsd4
\cbsd4\CBSD
\cbsd4\CBSD\cbsky4
\cbsd4\CBSD\cbzody6
\cbsd4\CBSD\Color_cor
\cbsd4\CBSD\RSR
\cbsd4\CBSD\SKY4Data
\cbsd4\CBSD\volemis
\cbsd4\CBSD\zodydata
\cbsd4\CBSD_SRC
\cbsd4\CBSD_SRC\cbsky4
\cbsd4\CBSD_SRC\cbzody6
\cbsd4\CBSD_SRC\fitsio
\cbsd4\CBSD_SRC\sky4data
\cbsd4\CBSD_SRC\VolEmis
\cbsd4\dataout
\cbsd4\dataout\cbsky4
\cbsd4\dataout\cbzody6
```

2.3 Unix/Linux/Irix Installation

The Unix tar file expands to 278 Megabytes. The unix does not include executables or unformatted data. All files must be built. We recommend two parallel directory structures, one for the source code and ASCII version of data files, and one for the executables. This minimizes the number of files to track and allows the deletion of the entire source directory tree when the executables and data files are built. If the tar extraction utility is used, this structure will be created automatically. A manual installation requires the user to create the directory structure (see the Windows section for details on the structure).

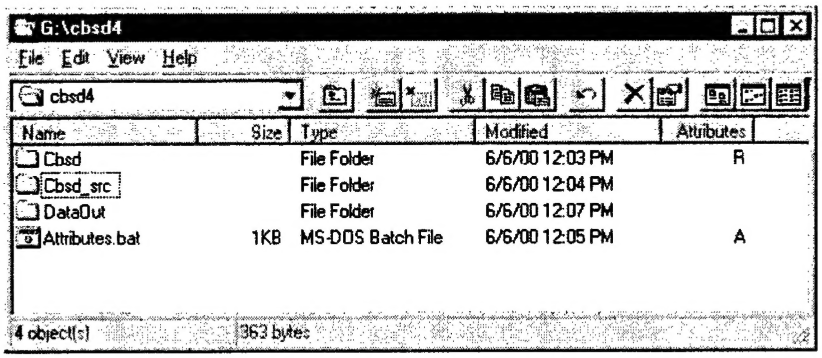
2.4 Visual Fortran based install (Windows NT)

To install the CBSD package, simply copy the files on the CD to your hard drive:

1. Create a new folder at the top level of your hard drive.
2. Rename the folder from New Folder to cbsd4.
3. Copy from the CD cbsd4 folder to the hard drive cbsd4 folder the following folders and files:
   Cbsd
   Cbsd_src
   Attributes.bat
   Note: You do not need the unix_tar folder or the Docs folder to run the software.

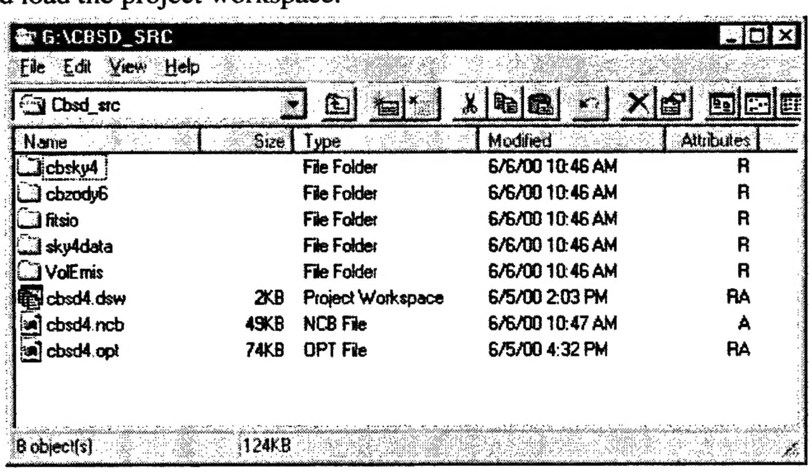
The Files on CDs are stored as read only. You need to remove the read-only attribute (under DOS/Windows) before making changes to files. The batch file "Attributes.bat" located in the \cbsd4\ directory will do this.

4. Double click the Attributes.bat icon in the folder.



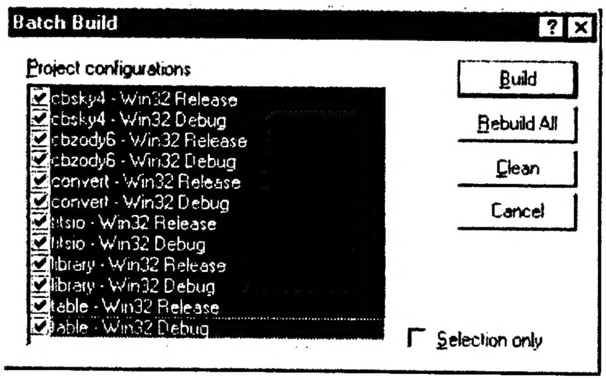
All source codes are under a single Compaq Visual Fortran workspace project named CBSD4.dsw. There are six projects in the workspace: Cbsky4, Cbzody6, Convert, Fitsio, Library, and Table. Library, Table, and Convert take the ASCII formatted datasets and convert them to unformatted, binary data for reduced time to load the files. The Fitsio project builds the NASA/GSFC release of the FITS (Flexible Image Transport System) routine library. The celestial model codes are in the Cbsky4 and Cbzody6 projects. **A makefile, fitsio.mak, is also supplied, for users who do not have Visual Fortran.**

5. Double-click on the CBSD4.dsw icon, or select Visual Fortran's Developer Studio from the Windows Start Menu and load the project workspace.



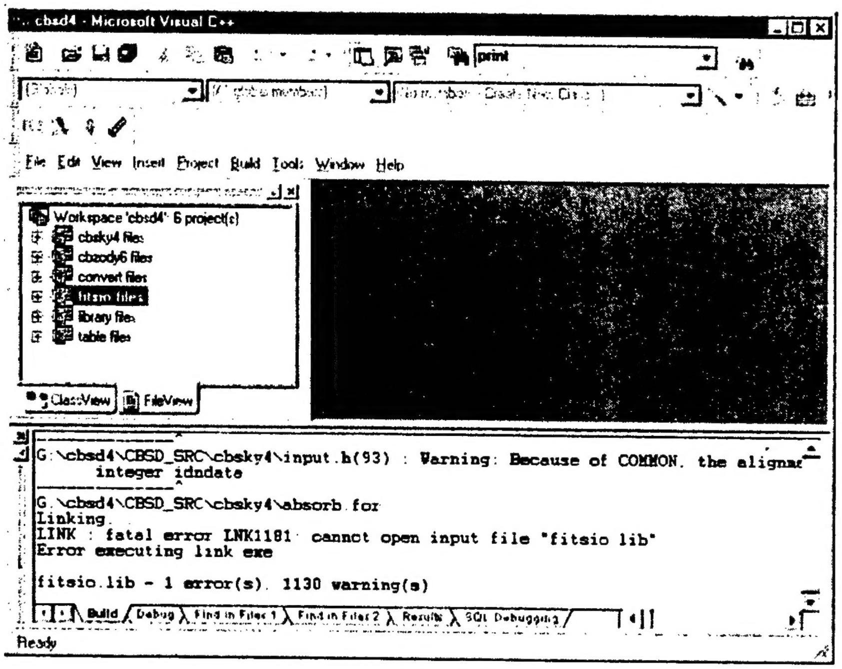
Before building the programs, clean up the debug and release directories.

6. Select the **Build|Batch Build**… menu item and select all the projects.
7. Click any of the checkboxes and all will be checked.
8. Then click "Clean."

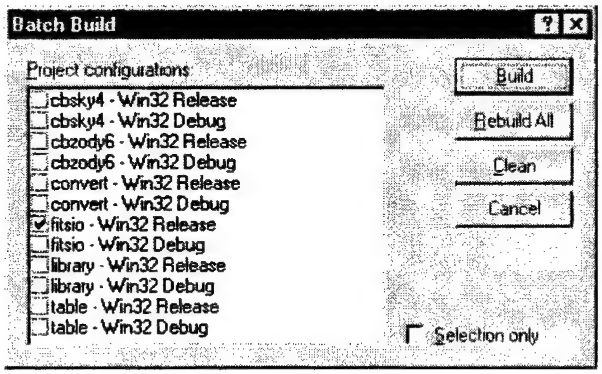


Now you are ready to build the programs. First, build the FITS library (fitsio.lib) from the project workspace. Table, library, and convert do not require the fitsio library. However, CBSKY4 and CBZODY6 require the fitsio library.

9. Right-click the fitsio project in the project workspace FileView pane, and select "Set as Active Project."



10. Select the **Build|Batch Build**… menu item.
11. Check the fitsio-Win32 Release option in the **Batch Build** Dialog Box.
12. Click Build.



The remaining codes (cbsky4 and cbzody) can now be built.

13. Set the CBSKY4 project as the Active Project. (Right click the project cbsky4 in the FileView pane of the cbsd4 project workspace and select Active Project from the pop-up menu.)
14. Select the **Project|Settings...** menu option. At the **Debug** Tab, in the **General** section, look at the text box labeled **Executable** for debug session. Verify that it is:

    \cbsd4\cbsd\cbsky4\cbsky4.exe

15. and if not, change the value. Likewise, on the Link tab, change the Output Filename in the General section.
16. Select the Build|Batch Build... menu item.
17. Check the cbsky4-Win32 Release option in the Batch Build Dialog Box.
18. Click Build.

    The cbsky4 program is now built.

19. Set the CBZODY6 project as the Active Project.
20. Select the Project|Settings... menu option. At the Debug Tab, in the General Section, look at the text box labeled Executable for debug session. Verify that it is:

    \cbsd4\cbsd\cbzody6\cbzody6.exe

21. and if not, change the value. Likewise, on the Link tab, change the Output Filename in the General section.
22. Select the Build|Batch Build... menu item.
23. Check the cbzody6-Win32 Release option in the Batch Build Dialog Box.
24. Click Build.

    The cbzody6 program is now built.

In order to run the cbsky4 and cbzody6 programs, the input data files need to be converted from ASCII or text format to binary format. This allows optimal data access and improves run times. The programs are delivered with the executables responsible for data conversion, so you can simply double click the program icon, or if you prefer, you can build them and run them from the project workspace. The following three programs should be run to perform the file conversion:

\cbsd4\cbsd\sky4data\table.exe

\cbsd4\cbsd\sky4data\library.exe

\cbsd4\cbsd\volemis\convert.exe

The table.exe program creates the data file halo.dat (1,608 bytes). The library.exe program creates the galaxy_library.dat (5,328 bytes) and the library.dat (115,884 bytes) files. The convert.exe program creates the sil35gip.bin (792,008 bytes) file.

To build and run the convert program from the project workspace:

25. Right-click the convert project in the FileView pane, and choose "Set as Active Project" from the pop-up menu.
26. Right-click the convert project in the FileView pane, and choose "Build" from the pop-up menu.
27. Press F5 (Function Key 5) to run the program.
28. Verify that the file \cbsd4\cbsd\volemis\sil35gip.bin exists, was created on this date/time, and is 792,008 bytes.

To build and run the **table** program from the project workspace:

29. Right-click the **table** project in the FileView pane, and choose "Set as Active Project" from the pop-up menu.
30. Right-click the **table** project in the FileView pane, and choose "Build" from the pop-up menu.
31. Press F5 (Function Key 5) to run the program.
32. Verify that the file \cbsd4\cbsd\sky4data\halo.dat exists, was created on this date/time, and is 1,608 bytes.

To build and run the **library** program from the project workspace:

33. Right-click the **library** project in the FileView pane, and choose "Set as Active Project" from the pop-up menu.
34. Right-click the **library** project in the FileView pane, and choose "Build" from the pop-up menu.
35. Press F5 (Function Key 5) to run the program.
36. Verify that the file \cbsd4\cbsd\sky4data\galaxy_library.dat exists, was created on this date/time, and is 5,328 bytes.
37. Verify that the file \cbsd4\cbsd\sky4data\library.dat exists, was created on this date/time, and is 115,884 bytes.

Finally, the CBSD codes are ready to run. Make sure the input files are modified to match your installation following the guidelines of Sections 3 and 4. You can skip Section 2.5, unless you plan to run CBSD on a Linux or Unix operating system.

2.5 Unix/Linux/Irix based install

The Unix distribution has been tested on a SGI O2 Unix system and a RedHat 6.2 linux system running on an Intel Pentium computer. Changes may be necessary for other platforms. The Unix/Linux distribution is provided in two gziped tar files named cbsd.tar.gz and cbsd4_src.tar.gz. Ungzip and untar the files using:

```
gzip –d cbsd.tar.gz
tar xvf cbsd.tar
gzip –d cbsd4_src.tar.gz
tar xvf cbsd4_src.tar
```

Note: On some Irix systems, the files appear as Cbsd4~1.gz and Cbsdta~1.gz. Uncompress them using the "gzip -d" command to get Cbsd4~1 and Cbsdta~1 then untar those files using "tar xvf Cbsd4~1" and "tar xvf Cbsdta~1".

Two directory structures are created, cbsd and cbsd4_src. You should also create a separate directory, dataout, with the two subdirectories dataout/cbsky4 and dataout/cbzody6, for the programs to put the output files.

The directory tree cbsd4_src has five subdirectories:

cbsd4_src/fitsio,

cbsd4_src/cbsky4,

cbsd4_src/cbzody6,

cbsd4_src/sky4data, and

cbsd4_src/volemis.

There is a Makefile file included in each source directory. The Makefiles will require some editing for your system.

The programs cbsky4 and cbzody6 link to the fitsio library. Thus, the first item to build is the libfitsio.a library. In the cbsd4_src/fitsio directory you must execute either configure or ./configure, depending upon operating system, to build the Makefile.

After configuring your system it may be necessary to modify the Makefile file. Check the FITSIO documentation if necessary. Then issue MAKE which will build the library libfitsio.a.

In each of the remaining source directories first edit the Makefile and change the HOME parameter to the directory containing the cbsd directory. Next issue MAKE in each of the directories. Check the trouble shooting section (Section 5) if you run into problems building these programs.

- cbsky4
- cbzody6
- sky4data
- volemis

Before running the celestial simulation models, you need to create some binary data files from the ASCII files of the distribution. Run the following programs.

cbsd/sky4data/table,

cbsd/sky4data/library, and

cbsd/volemis/convert.

The codes cbsd/cbzody6 and cbsd/cbsky4 are ready to run once the input files are modified. To run cbsky4 open the cbsky4.inp (all lower case) file, and edit the following lines (Note: the semi-colon is a comment delimiter):

```
[path]
architecture = UNIX
path = /usr/YOU/dataout/cbsky4  ;The path for the output files must exist
data_path = /usr/YOU/cbsd/sky4data   ;The location of the data files
code_path = /usr/YOU/cbsd/cbsky4     ;The location of the executable

[spectral]
start_wavelength = /usr/YOU/cbsd/rsr/msx_a.txt ;The rsr filespec
```

Then, run the cbsky4 program. The output data can be viewed in the dataout/cbsky4 directory.

To run cbzody6 open the cbzody.inp (all lower case) file, and edit the following lines (Note: the semi-colon is a comment delimiter):

```
[path]
architecture = UNIX
path = /usr/YOU/dataout/cbzody6              ; directory for outputs to go
code_path = /usr/YOU/\cbsd/cbzody6           ; executable directory
vol_emiss_path = /usr/YOU/cbsd/volemis       ; volemis directory
data_path = /usr/YOU/cbsd/zodydata           ; zodydata directory

...
[spectral]
...
start_wavelength = /usr/YOU/rsr/DIRBE05.DAT  ;rsr filter function filespec
```

Then, run the cbzody6 program. The output data can be viewed in the dataout/cbzody6 directory.

2.6 Other installs

CBSD4 has not been extensively tested on different platforms so the extent of the changes needed for other platforms is not known.

General requirements:

- Use bytes as the length for the RECL= parameter in open statements.
- Modify the imopen routine for your compiler
- Modify the make files for your compiler.

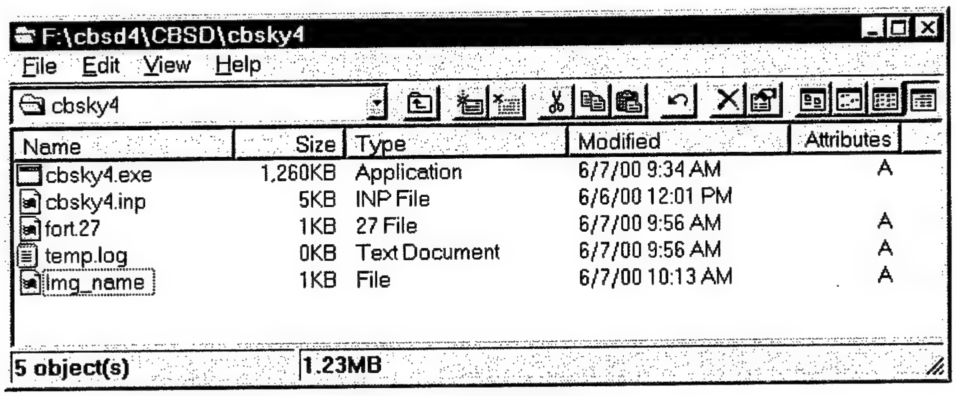
3 The CBSKY4 Model

3.1 Introduction

The CBSKY4 module generates images that contain stellar point sources. This includes not only the stars but also includes point source galaxies, HII regions, and planetary nebulae.

3.2 Using CBSKY4 For the First Time

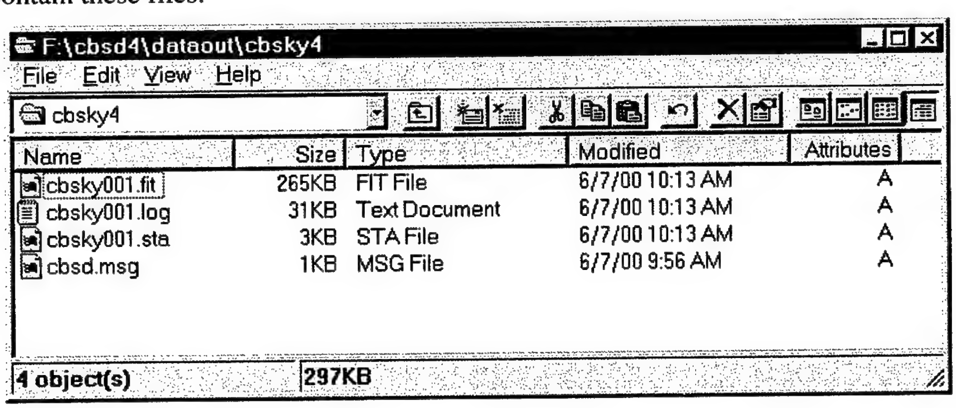
To run the cbsky4 program on a Windows system, go to the folder \cbsd4\cbsd\cbsky4, and double-click the cbsky4 icon. The program reads the cbsky4.inp text file to get information defining the simulation image (e.g., the band-pass, the central Right Ascension and Declination, the IFOV of a pixel, the image array dimensions, etc.), settings for model options (e.g., magnitude limit for modeling), and system path settings (e.g., where the sky4data folder is, where to put the output). When the program completes, there will be three more files in the folder: fort27, temp.log, and Img_name.



The Img_name file is a text file containing a single line:

\CBSD4\dataout\cbsky4\cbsky001.fit

which specifies the filename and location of the output image file. As delivered, the cbsky4.inp file specifies that outputs are to be written in the \CBSD4\dataout\cbsky4 folder. At the end of this run the folder will contain these files:

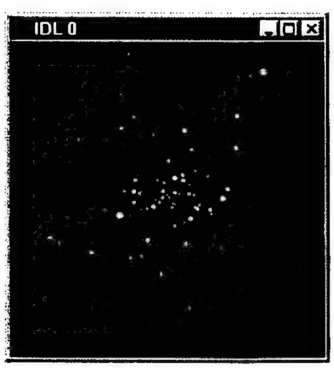


The files are stored with an index value, eg., cbsky(*nnn*).sta, where *nnn* is incremented each time the code is run. To start reindexing, define a new path for the output files, or delete the output from this folder.

The cbsky001.fit file is the image file in FITS (Flexible Image Transport System) format. If you do not have a fits reader, the file has a text header which can be viewed in a text editor (just open the fits file in a text editor). The following is the header for the file generated by the installation cbsky4.inp.

```
SIMPLE  =                    T / file does conform to FITS standard
BITPIX  =                  -32 / number of bits per data pixel
NAXIS   =                    2 / number of data axes
NAXIS1  =                  256 / length of data axis 1
NAXIS2  =                  256 / length of data axis 2
COMMENT   FITS (Flexible Image Transport System) format defined in Astronomy and
COMMENT   Astrophysics Supplement Series v44/p363, v44/p371, v73/p359, v73/p365.
COMMENT   Contact the NASA Science Office of Standards and Technology for the
COMMENT   FITS Definition document #100 and other FITS information.
BZERO   =         0.000000E+00 / Intensity of 0 byte pixels
BSCALE  =             1.000000 / Intensity Scale Factor
BUNIT   = ' W/cm2/sr'          / Intensity Unit
CTYPE1  = '   Long.(Deg)'      / X-axis
CRPIX1  =                  1.0 / Pixel Origin
CRVAL1  =           208.000000 / Xzero
CDELT1  =            -1.000000 / Xscale
CROTA1  =                  0.0 / No Rotation
CTYPE2  = '   Lat. (Deg)'      / Y-axis
CRPIX2  =                  1.0 / Pixel Origin
CRVAL2  =          -128.000000 / Yzero
CDELT2  =             1.000000 / Yscale
CROTA2  =                  0.0 / No Rotation
DATAMIN =         0.000000E+00 / Minimum Intensity
DATAMAX =         4.732501E-08 / Maximum Intensity
DATE    = '07/06/00'           / FITS file creation date (dd/mm/yy)
ORIGIN  = 'PIP-MRC/NH'         / FITS file creation person-organization
INSTRUME= 'CBSD    '           / Model
HISTORY   CODE     = CBSKY4
HISTORY   RELEASE = 1.07
AUTHOR  = 'Noah    '           / Code developer
DATE-OBS= ' 7/ 3/2000'         / UTC (dd/mm/yyyy)
HISTORY   TITLE    = CBSKY4
HISTORY   PROJECTION = Gnomonic
HISTORY   COORDINATE_SYSTEM = Galactic
HISTORY   HDRTYPE = HDR0       / 4-Byte Real; I=Image Value
HISTORY   KWAVE1   =  5.49     / Starting Wavelength in microns
HISTORY   KWAVE2   = 11.76     / Ending Wavelength in microns
HISTORY   KJUL= 2451610.500000 / Julian Date
END
```

For detailed information about FITS formats, consult the standard documents produced by NASA. Following the header is the binary image. The output image is 256 columns by 256 rows. Each pixel is a 4-byte floating point number. Here is the image displayed in IDL with log10 scaling, and stretched for minimum value of -13 and maximum value of -9.



To change the image parameters, edit the cbsky4.inp file using a text editor. Section 3.3 describes the parameters and lists their allowed values.

3.3 Creating CBSKY4 Inputs

The CBSKY4 input file (cbsky4.inp) follows the standard Microsoft Windows© ini format described in Section 1.5. It must reside in the folder containing cbsky4.exe. The cbsky4.inp file can be edited using any text editor.

Here is a listing of a sample cbsky4.inp file (Note: In the input file the semi-colon (;) is treated as a comment delimiter.):

```
[path]
architecture = DOS
path = d:\dataout\cbsky4        ;The path for the output files must exist
data_path = d:\cbsd\sky4data    ;The location of the data files
code_path = d:\cbsd\cbsky4      ;The location of the executable
verbose = no

[cbsky4]
real_stars = yes
statistical_stars = yes
clouds = yes
magnitude_limit = 12.0
seed = 346
method = center
catalog = no
nodesfile = node_iah.dat
elementsfile = elem_iah.dat
count_statistices = no
map = no
errmap = no
extmap = no
extinction = no

[image]
image = yes
x_column_pixels = 256
y_column_pixels = 256
pixel_size = 0.1
image_projection = GNOMONIC
image_center_longitude_degrees = 90.
image_center_latitude = 0.0
units = W/CM2/MICRON
image_type = 4-BYTE REAL
output_format = FITS

[positional]
observer_altitude = 2026.
observer_geographic_latitude = 31.9576
observer_geographic_longitude = 111.5947
reference_frame = j2000
positions = apparent
reference_system = geocentric
coordinate_system = galactic

[convolution]
convolution = yes
point_spread_function = gaussian
psf_half_width = 1.01

[spectral]
start_wavelength = d:\cbsd\rsr\msx_a.txt
delta_wavelength = 0.1
band_center = 8.276
bandwidth = 3.362

[time]
observation_date = 7 3 2000
observation_time = 0 0 0.0
```

Following the ini conventions, there are section headings (e.g., "path") and keywords (e.g. "path") which must be listed in the file.

```
[path]
path = \CBSD4\dataout\cbsky4    ;The path for the output files must exist
code_path = \cbsd4\cbsd\cbsky4 ;The location of the executable
```

The user can modify the value of the keyword, and below are listed the allowed values for each keyword (parameter).

The output files are stored in the [path] path keyword. Note that the input file is not saved with the output files. So, it is generally a good idea to copy the cbsky4.inp file to the output directory, using the indexing convention of:

Cbsky(*nnn*).inp

Where nnn is the run number starting with 001 and incrementing (002, 003, ...) each time the code is executed. After saving your last inputs, you can start editing the file.

The file cbsky4.inp consists of these seven main sections:

38. **[path]**
39. **[cbsky4]**
40. **[image]**
41. **[spectral]**
42. **[convolution]**
43. **[time]**
44. **[positional]**

Each section has one or more keywords that require either a named list option or numeric input. For example of named list options, for:

```
[image]
image_projection = Rectangular
```

the image_projection keyword can take on any of the following values: rectangular, gnomonic, or mollweide. Likewise the coordinate_system keyword in the positional section:

```
[positional]
coordinate_system = Equatorial
```

can take on any of the following values: equatorial, galactic, or ecliptic. Capitalization doesn't matter but spelling counts, so be careful.

For the numeric entries, you define the value. The tables below list the value type (e.g., integer, float) and the parameter descriptions tell the range of allowed values and their units. For example:

```
[positional]
Observer_geographic_longitude = -34.5
```

specifies that the observer coordinate is located at -34.5 degrees West of Greenwich. The possible range of values is -180° to +180° and for all CBSD codes the Positive West convention is used.

Keywords, descriptions, and allowed values are provided on Pages 15-42

[path]

Parameter	Usage	Definition
architecture		
	DOS	Flag for MS-DOS and Windows based computers.
	UNIX	Flag for Unix and Linux systems.
path		
	"Path"	Full path to the subdirectory where the CBSKY image data are to be stored.
code_path		
	"Path"	Full path to the executables and user input files.
data_path		
	"Path"	Full path to the subdirectory where the SKY statistical stellar database is stored.
verbose		
	No	Does not write additional information to log file.
	Yes	Writes additional information to the log file to aid in debugging.

[cbsky4]

Parameter	Usage	Definition
map		
	No	Future usage. Just say No.
method		
	Center	Use the center point of the finite element triangle to obtain the stellar density distribution.
	Vertices	Use the vertices of the finite element triangle and average the three values to obtain the stellar density distribution.
log_output		
	"Filename"	An optional name for the output file. If none is specified, the default of CBSKYxxx.LOG is used. where xxx is a running numerical index starting at 000. The index number is the same as the one generated by the ***image*** parameter.
magnitude_limit		
	Floating Point	The limiting magnitude of the image generation. An increased magnitude limit means increased fidelity but also increased run time.
catalog		
	No	Does not generate a file with the list of stars used in the image.
	Yes	Generates a file name of the form CBSKYxxx.CAT where xxx is a running numerical index starting at 000. The index number is the same as the one generated by the ***image*** parameter. The form of the file is three ASCII columns. The first column is the Right Ascension in degrees. The second column is Declination in degrees. The third column is the flux with the units given by the ***units*** parameter.
catalog_limit		
	Floating Point	A magnitude value to indicate the limiting magnitude of the ASCII catalog. This value can be less than or equal to the ***magnitude_limit*** parameter.
count_statistics		
	No	Do not generate count statistics information.
	Yes	Generate the count statistics information. The count statistics are used to make the *log(N) - log(S)* plots. A value of *Yes* creates a file called CBSKYxxx.STA which is eleven columns of values; magnitude bin number, stars per bin, cumulative stars per bin, magnitude, log N, magnitude, log N per magnitude, log flux, log N, log flux, log N per magnitude.

[cbsky4] cont.

Parameter	Usage	Definition
real_stars		
	No	Do not produce an image using the real star database.
	Yes	Produce an image using the real star database files which consist of the Yale BSC in the visual and a merged IRAS/MSX catalog at 12 microns. These stars all have a spectral template applied and will be visible (possibly) in all wavelengths.
statistical_stars		
	No	Do not produce an image with statistical stars.
	Yes	Produce an image populated with statistical stars. These stars do not produce an image with the correct placement of stars compared to catalog images. However, the number of stars per magnitude interval per spectral type is correct.

[cbsky4] cont.

Parameter	Usage	Definition
clouds		
	No	Do not produce an image with the molecular cloud enhancements.
	Yes	A separate component of the statistical stars are the molecular cloud stars. These are regions with higher than average stellar densities. A *Yes* indicates that the star clouds will be included. The star clouds are treated as ellipses with a Gaussian fall off in number density. Examples of star clouds are the LMC and Orion.
nodesfile		
	"Filename"	The entire sky of CBSKY4 was divided, by finite element grid generation, into a set of triangles. The smaller triangles represent higher stellar density regions. The ***nodesfile*** represents the name of the file (without path information) with the coordinates of the vertices of the triangles. Three node files exist for different spectral regions and resolutions. *node_iah.dat* for high resolution infrared, *node_ia.dat* for low resolution infrared, and *node_va.dat* for visible wavelengths. This file must be paired with ***elementsfile***.
elementsfile		
	"Filename"	The entire sky of CBSKY4 was divided, by finite element grid generation, into a set of triangles. The smaller triangles represent higher stellar density regions. The ***elementsfile*** represents the name of the file (without path information) with the connection paths between nodes. Three element files exist for different spectral regions and resolutions. *elem_iah.dat* for high resolution infrared, *elem_ia.dat* for low resolution infrared, and *elem_va.dat* for visible wavelengths. This file must be paired with ***nodesfile***.
seed		
	Integer	A random number seed value
extinction		
	No	Future usage, leave as *No*.
errmap		
	No	Future usage, leave as *No*.
extmap		
	No	Future usage, leave as *No*.

[image]

Parameter	Usage	Description
image		
	No	Suppress image generation (for debugging only).
	Yes	Create an image of the form CBSKYxxx.yyy where xxx is a running numerical index starting at 000. The yyy extension indicates type of image or the header. If yyy is BIN the image is headerless. If the extension if FIT the file is in FITS format. See the related parameters ***output_format*** and ***image_type***.
x_column_pixels		
	Integer	The horizontal image size in pixels. Valid range is 1 to 4096.
y_row_pixels		
	Integer	The vertical image size in pixels. Valid range is 1 to 4096.
pixel_size		
	Floating point	The angular size of each pixel in degrees per pixel. Each pixel is taken to be square.
image_center_longitude_hours		
	Floating point.	The ***image_center_longitude_hours*** parameter specifies the longitude center of the image in decimal hours. For the equatorial system, ***image_center_longitude*** is Right Ascension. The range of values is from 0.0^h to 24.0^h.
	Integer (Hours), Floating Point (Minutes).	Optionally, the ***image_center_longitude_hours*** parameter can specify the longitude center of the image in integer hours and decimal minutes. The range of values is from $0^h\ 0.0^m$ to $24^h\ 0.0^m$.
	Integer (Hours), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_longitude_hours*** parameter can specify the longitude center of the image in integer hours, integer minutes, and decimal seconds. The range of values is from $0^h\ 0^m\ 0.0^s$ to $24^h\ 0^m\ 0.0^s$.

[image] cont.

Parameter	Usage	Definition
image_center_longitude_degrees		
	Floating point.	The ***image_center_longitude_degrees*** parameter specifies the longitude center of the image in decimal degrees. For the equatorial system, ***image_center_longitude*** is Right Ascension which is formally measured in hours. Degrees are obtained by multiplying the Right Ascension, in hours, by 15. The range of values is from 0.0° to 360.0°.
	Integer (Degrees), Floating Point (Minutes).	Optionally, the ***image_center_longitude_degrees*** parameter can specify the longitude center of the image in integer degrees and decimal minutes. The range of values is from 0° 0.0' to 360° 0.0'.
	Integer (Degrees), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_longitude_degrees*** parameter can specify the longitude center of the image in integer degrees, integer minutes, and decimal seconds. The range of values is from 0° 0' 0.0" to 360° 0' 0.0".
image_center_latitude		
	Floating point.	The ***image_center_latitude*** parameter specifies the latitude center of the image in decimal degrees. For the equatorial system, ***image_center_latitude*** is Declination. The range of values is from -90.0° to +90.0°.
	Integer (Degrees), Floating Point (Minutes).	Optionally, the ***image_center_latitude*** parameter can specify the latitude center of the image in integer degrees and decimal minutes. The range of values is from -90° 0.0' to +90° 0.0'.
	Integer (Degrees), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_latitude*** parameter can specify the longitude center of the image in integer degrees, integer minutes, and decimal seconds. The range of values is from -90° 0' 0.0" to +90° 0' 0.0".
output_format		
	None	The *None* option produces an unformatted image, either byte or word as specified by ***image_type***, with no header.
	FITS	The *FITS* format is standard FITS specified by NASA GSFC FITS Committee. A copy of "Implementation of the Flexible Image Transport System (FITS)", NOST Report 100-03b is included with this documentation.

[image] cont.

Parameter	Usage	Definition
units		
	Jy/sr	Flux unit of Janskys/steradian.
	Mjy/sr	Flux unit of MegaJanskys/steradian.
	Jy	Flux unit of Janskys.
	MJy	Flux unit of MegaJanskys.
	W/cm2/sr/um	Flux unit of $W/cm^2/sr/\mu m$.
	W/cm2/sr	Flux unit of $W/cm^2/sr$.
	W/cm2	Flux unit of W/cm^2.
	W/cm2/um	Flux unit of $W/cm^2/\mu m$.
image_type		
	Byte	The output image is stored as an 8-bit, unformatted image.
	4-byte-real	The output image is stored as a 4-byte, unformatted, floating point number. Floating point data are stored in the native format of the computer used to generate images. When transferring the image to a UNIX platform, the user must be aware of the need to perform byte swapping.
image_projection		
	Rectangular	The *Rectangular* projection is the familiar Mercator type of projection with parallel lines of latitude and parallel lines of longitude. Maps produced by CBSKY using the *Rectangular* projection do not "wrap" at the poles. This is not the case at 0^h and 24^h were "wrapping" is allowed.
	Gnomonic	The *Gnomonic* projection is the projection of a sphere onto a plane from a tangent point; in other words, as a camera would see it.
	Mollweide	The *Mollweide* projection is an equal-area sinusoidal projection with the lines of latitude all parallel. The *Mollweide* projection is only used as an all sky projection. It is recommended that the ratio of ***x_column_pixels*** to ***y_row_pixels*** be 2:1 for a more pleasing image.

[positional]

The **[positional]** section specifies geometric parameters.

Parameter	**Usage**	**Definition**
observer_altitude		
	Floating point	The altitude, in meters, of an observer as defined by the Earth's reference geoid of 1976.
observer_geographic_latitude		
	Floating point	The observer coordinates in degrees referenced to the equator with positive north of the equator and negative south of the equator.
observer_geographic_longitude		
	Floating point	The observer coordinates in degrees relative to Greenwich with positive longitudes taken to be west of Greenwich and negative longitudes east of Greenwich. The possible range is -180° to +180°.
reference_frame		
	J2000	Image coordinates are specified as J2000.
	B1950	Image coordinates are specified as B1950.
	True	Image coordinates are specified as reference to the mean equinox and epoch of the observation.
coordinate_system		
	Equatorial	The equatorial system is the familiar Right Ascension and Declination coordinate system. The sides of the output image are always parallel to the lines of longitude and latitude, that is, no image rotations are possible.
	Galactic	A coordinate system based on the galactic coordinate l_{II} and b_{II} of B1950.
	Ecliptic	The ecliptic coordinate system.
positions		
	Apparent	Apparent positions are those with the ***reference_frame*** given as *True*, the mean equinox and epoch of the observation.
	Astrometric	Astrometric positions are referenced to a standard such as J2000, B1950, B1900, etc.
	Geometric	Geometric coordinates are positions of objects without light time corrections or aberrations.

[positional] cont.

Parameter	Usage	Definition
reference_system		
	Geocentric	A reference system centered on the Earth's center.
	Topocentric	A reference system centered on the position of the observer.
	Heliocentric	A reference system centered on the center of the Sun.
	Barycentric	A reference system centered on the solar system's barycenter.

[convolution]

The **[convolution]** section allows the optional definition of a point spread function.

Parameter	Usage	Definition
point_spread_function		
	No	Do not apply a point spread function.
	Gaussian	Apply a Gaussian point spread function with width given by ***gaussian_half_width***.
gaussian_half_width		
	Floating point	The half width at half maximum of the Gaussian point spread function in pixels. Ignored if ***point_spread_function***=*no*.

[spectral]

The **[spectral]** section has four parameters that specify the spectral characteristics of the image. CBSKY4 has a wavelength coverage of 0.2µm to 30µm.

Parameter	Usage	Definition
start_wavelength		
	"Filename"	The full path to a filter response file. The data are two columns, wavelength in microns and response. The data do not need to be evenly spaced. A cubic spline is used to interpolate the spectral filter. To aid in the usage of the cubic spline, the first and last values in the response file should be zero (0.0). The ***end_wavelength*** parameter is not used.
	Floating point	The start wavelength, in micrometers, of the bandpass.
	B	Use the standard B photometric band (0.44µm). The ***end_wavelength*** parameter is not used.
	V	Use the standard V photometric band (0.55µm). The ***end_wavelength*** parameter is not used.
	J	Use the standard J photometric band (1.25µm). The ***end_wavelength*** parameter is not used.
	H	Use the standard H photometric band (1.65µm). The ***end_wavelength*** parameter is not used.
	K	Use the standard K photometric band (2.22µm). The ***end_wavelength*** parameter is not used.
	2.4um	Use the COBE/DIRBE 2.4µm band (Band 2). The ***end_wavelength*** parameter is not used.
	12um	Use the IRAS 12µm band (Band 1). The ***end_wavelength*** parameter is not used.
	25um	Use the IRAS 25µm band (Band 2). The ***end_wavelength*** parameter is not used.
	1565A	Use the TD1 1565Å band. The ***end_wavelength*** parameter is not used.
	1400A	Use the TD1 1400Å band. The ***end_wavelength*** parameter is not used.
	1660A	Use the TD1 1660Å band. The ***end_wavelength*** parameter is not used.
end_wavelength		
	Floating point	The stop wavelength, in micrometers, of the bandpass.
delta_wavelength		
	Floating point	The ***delta_wavelength*** parameter specifies the step in wavelength, in micrometers, for an output spectrum.

[spectral] cont.

Parameter	Usage	Definition
flux_mag_zero		
	Floating point	An optional parameter that allows the in-band magnitude of objects to be quoted.
band_center		
	Floating point	If not specified the CBSD code will define the band center by a simple center of gravity calculation. This is not always in agreement with system developer's definitions. This parameter allows an override of the internal routines.
band_width		
	Floating point	If not specified, the CBSD code will define the bandwidth by a simple center of gravity calculation. This is not always in agreement with system developer's definitions. This parameter allows an override of the internal routines.

[time]

The **[time]** section specifies the date and time of the observation.

Parameter	Usage	Definition
observation_date		
	Integer (Day), Integer (Month), Integer (Year).	The ***observation_date*** is three numbers in the order integer day, numbered month, and 4-digit year. For example, 3 15 1995
	Floating Point	A second option exists for ***observation_date*** which can read the Julian date. Enter the complete Julian date, for example, 2445345.125 In this case the ***observation_time*** is not read.
observation_time		
	Integer (Hour), Integer (Minute), Floating Point (Seconds).	The ***observation_time*** parameter is the integer hour, integer minute and floating point second of the observation. For example, 12 23 55.06 Ignored if the ***observation_date*** is specified as a Julian date.
time_stamp		
	UT1	The time of observation is defined as the Mean Solar Time.
	UTC	The time of observation is defined as the Coordinated Universal Time - clock time referenced to Greenwich.
	TDT	The time of observation is defined as the Terrestrial Dynamical Time - Used for calculations of ephemerides.

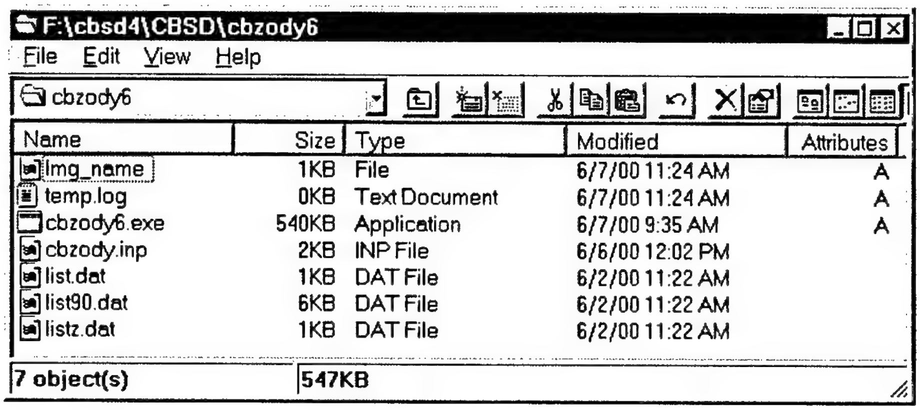
4 The CBZODY6 Model

4.1 Introduction

The CBZODY6 model gives the flux from the solar system dust cloud that is defined as the zodiacal light and the zodiacal bands.

4.2 Using CBZODY6 For the First Time

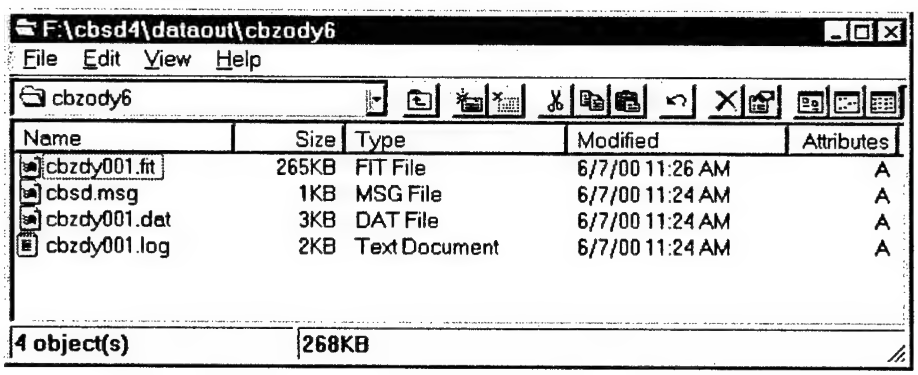
To run the cbzody6 program on a Windows system, go to the folder \cbsd4\cbsd\cbzody6, and double-click the cbzody6 icon. The program reads the cbzody.inp text file to get information defining the simulation image (e.g., the band-pass, the central Right Ascension and Declination, the IFOV of a pixel, the image array dimensions, etc.), settings for model options (e.g., scattering phase function for modeling), and system path settings (e.g., where the zodydata folder is, where to put the output). When the program finishes, there will be three more files in the folder.



The Img_name file is a text file containing a single line:

\CBSD4\dataout\cbzody6\cbzody001.fit

which specifies the filename and location of the output image file. As delivered, the cbzody6.inp file specifies that outputs are to be written in the \CBSD4\dataout\ cbzody6 folder. At the end of this run the folder will contain these files:

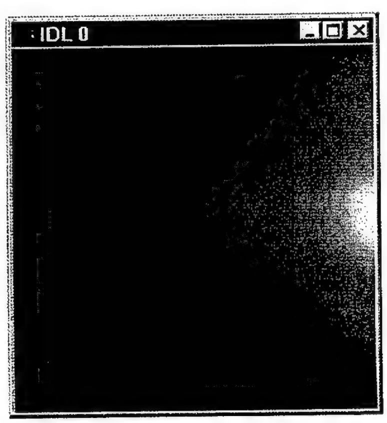


The files are stored with an index value, eg., cbzdy(*nnn*).dat, where *nnn* is incremented each time the code is run. To start reindexing, define a new path for the output files, or delete the output from this folder.

The cbzody001.fit file is the image file in FITS (Flexible Image Transport System) format. If you do not have a fits reader, the file has a text header which can be viewed in a text editor (just open the fits file in a text editor). The following is the header for the file generated by the installation cbzody.inp.

```
SIMPLE  =                    T / file does conform to FITS standard
BITPIX  =                  -32 / number of bits per data pixel
NAXIS   =                    2 / number of data axes
NAXIS1  =                  256 / length of data axis 1
NAXIS2  =                  256 / length of data axis 2
COMMENT   FITS (Flexible Image Transport System) format defined in Astronomy and
COMMENT   Astrophysics Supplement Series v44/p363, v44/p371, v73/p359, v73/p365.
COMMENT   Contact the NASA Science Office of Standards and Technology for the
COMMENT   FITS Definition document #100 and other FITS information.
BZERO   =         0.000000E+00 / Intensity of 0 byte pixels
BSCALE  =             1.000000 / Intensity Scale Factor
BUNIT   = '   Jansky/sr'       / Intensity Unit
CTYPE1  = '   Long.(Deg)'      / X-axis
CRPIX1  =                  1.0 / Pixel Origin
CRVAL1  =           102.800003 / Xzero
CDELT1  =            -0.100000 / Xscale
CROTA1  =                  0.0 / No Rotation
CTYPE2  = '   Lat. (Deg)'      / Y-axis
CRPIX2  =                  1.0 / Pixel Origin
CRVAL2  =           -12.800000 / Yzero
CDELT2  =             0.100000 / Yscale
CROTA2  =                  0.0 / No Rotation
DATAMIN =         3.719145E+07 / Minimum Intensity
DATAMAX =         1.065804E+08 / Maximum Intensity
DATE    = '07/06/00'           / FITS file creation date (dd/mm/yy)
RIGIN   = 'PIP-MRC/NH'         / FITS file creation person-organization
INSTRUME= 'CBSD     '          / Model
HISTORY   CODE     = CBSKY4
HISTORY   RELEASE = 6.06
AUTHOR  = 'Noah     '          / Code developer
DATE-OBS= '15/ 4/2000'         / UTC (dd/mm/yyyy)
HISTORY   TITLE    = CBZODY6
HISTORY   PROJECTION = Gnomonic
HISTORY   COORDINATE_SYSTEM = Ecliptic
HISTORY   HDRTYPE = HDR0       / 4-Byte Real; I=Image Value
HISTORY   KWAVE1  =   7.66     / Starting Wavelength in microns
HISTORY   KWAVE2  = 17.47      / Ending Wavelength in microns
HISTORY   KJUL= 2451649.500000 / Julian Date
END
```

For detailed information about FITS formats, consult the standard documents produced by NASA. Following the header is the binary image. The output image is 256 columns by 256 rows. Each pixel is a 4-byte floating point number.



To change the image parameters, edit the cbzody.inp file using a text editor. Section 4.3 describes the parameters and lists their allowed values.

4.3 Creating CBZODY Input Files

The cbzody input file (cbzody.inp) follows the standard Microsoft Windows © ini format described in Section 1.5. It must reside in the folder containing cbzody6.exe. The cbzody.inp file can be edited using any text editor. All inputs are read from a file called cbzody.inp except for the observational datasets, which require command line information.

Here is a listing of a sample cbzody.inp file (Note: In the input file the semi-colon (;) is treated as a comment delimiter.):

```
[path]
architecture = DOS     ;for UNIX use unix
path=c:\dataout\cbzody6\       ;location of output images
code_path = c:\cbsd\cbzody6    ;location of executables
data_path = c:\cbsd\zodydata   ;ancillary CBZODY6 data files
vol_emiss_path = c:\cbsd\volemis      ;location of volumetric data files
verbose = yes  ;lots of output

[cbzody]
cloud = yes    ;calculate the dust cloud
bands = yes    ;calculate the dust bands
ring = no      ;no ring calculation
model = gnomonic       ;pixel-by-pixel calculation
Volume_emissivity = sil35gip.bin      ;65 %astronomical silicate and 35% graphite
material_file = unformatted    ;binary version
Volume_scattering = silip.sct ; silicate scattering
Gain = no      ;for use with IRAS data only
cio = no       ;for use with COBE data only

[image]
x_column_pixels = 256  ;image size
y_row_pixels = 256
pixel_size = 0.5       ;per pixel FOV
image_center_latitude = 30.75 ;degrees
image_center_longitude_degrees = 203.53       ;degrees
image_projection = gnomonic
image_type = 4-byte real
output_format = FITS
units = Jy/sr  ;output units selection

[positional]
observer_altitude = 0.         ;observer altitude above MSL
observer_geographic_latitude = -23.0  ;geographic coordinates
observer_geographic_longitude = -120.0
reference_frame = J2000        ;J2000
positions = apparent   ;equinox and epoch of date
coordinates = equatorial       ; RA and Dec

[spectral]
start_wavelength = c:\cbsd\rsr\dirbe05.dat    ;dirbe band 5
delta_wavelength = 0.1                ;microns, must match spacing of volumetric files
band_center = 13.518   ;microns
bandwidth =6 .5599     ;microns
color_correction = 0.93        ;correct color

[time]
observation_date = 16 7 1994  ;dmy
observation_time = 6 54 0     ;hms
time_stamp = UTC       ;always use UTC
```

Following the ini conventions, there are section headings (e.g., "path") and keywords (e.g. "path") that must be listed in the file.

```
[path]
path = c:\dataout\cbzody6\      ;location of output images
code_path = c:\cbsd\cbzody6     ;location of executables
data_path = c:\cbsd\zodydata    ;ancillary CBZODY6 data files
```

The user can modify the value of the keyword, and below are listed the allowed values for each keyword (parameter).

The output files are stored in the [path] path keyword. Note that the input file is not saved with the output files. So, it is generally a good idea to copy the cbzody.inp file to the output directory, using the indexing convention of:

Cbzody(*nnn*).inp

Where nnn is the run number starting with 001 and incrementing (002, 003, ...) each time the code is executed. After saving your last inputs, you can start editing the file.

The input file is divided into six sections:

[path]	**[cbzody]**	**[image]**
[spectral]	**[positional]**	**[time]**

Each section specifies related parameters. The sections are similar for all of the CBSD codes with the exception of **[cbzody]** which are inputs specific to this model.

Each section has one or more keywords that require either a named list option or numeric input. For example of named list options, for:

```
[image]
image_projection = Rectangular
```

the image_projection keyword can take on any of the following values: rectangular, gnomonic, or mollweide. Likewise the coordinate_system keyword in the positional section:

```
[positional]
coordinate_system = Equatorial
```

can take on any of the following values: equatorial, galactic, or ecliptic. Capitalization doesn't matter but spelling counts, so be careful.

For the numeric entries, you define the value. The tables below list the value type (e.g., integer, float) and the parameter descriptions tell the range of allowed values and their units. For example:

```
[positional]
Observer_geographic_longitude = -34.5
```

Specifies that the observer coordinate is located at -34.5 degrees west of Greenwich. The possible range of values is -180° to +180° and for all cbsd codes, the positive west convention is used.

The following tables list the keywords, their descriptions, and their allowed values.

[path]

Parameter	Usage	Definition
architecture		
	DOS	Flag for MS-DOS computers.
	UNIX	Flag for UNIX systems.
path		
	"Path"	Full path to the subdirectory where the image data are to be stored.
vol_emiss_path		
	"Path"	Full path to the subdirectory where the CBZODY6 volume emissivity database is stored.
code_path		
	"Path"	Full path to the subdirectory where the CBZODY6 executable and user input are stored.
data_path		
	"Path"	Full path to the subdirectory where the SKY statistical stellar database is stored.
verbose		
	No	Does not write additional information to log file.
	Yes	Writes additional information to the log file to aid in debugging.

[cbzody]

Parameter	Usage	Definition
model		
	Area	Produce an image of size nxm where n and m are specified by the user calculating the radiance at every pixel used with a *Gnomonic* or *Rectangular* image projection
	Los	Produce an intensity along a single line of sight, also capable of reading a list.
		When ***model*** = *los*, five columns are output to CBZODY.LOG. The first two are the pointing direction in degrees for the chosen coordinate system (latitude then longitude). The next are the total (cloud + bands), cloud, and band intensities in MJy/sr.
	Spectrum	Produce a spectrum between two wavelengths.
		When ***model*** = *spectrum*, five columns are output to CBZODY.LOG. The first column is the wavelength in microns. The next are the total (cloud + bands), cloud, and band intensities in MJy/sr. The final is the Brightness temperature in Kelvin.
Band power law		
	Integer	Power law for the migrating band function. This parameter must correspond to input values in the cbaparms.dat file.
gain		
	No	Do not use the empirical gain function.
	Yes	The IRAS satellite was known to have time varying gain variations and gain variations with source intensity. These have been partially characterized and can be included in the calculations. At this point, the gain model should only be used when comparing directly with IRAS data.

Parameter	Usage	Definition
cloud		
	No	Specifies that the dust cloud structure is not to be included in the flux calculation.
	Yes	Specifies that the dust cloud structure is to be included in the flux calculation.
bands		
	No	Specifies that the band structure is not to be included in the flux calculation.
	Yes	Specifies that the band structure is to be included in the flux calculation.
ring		
	No	Do not use the solar system ring construct.
	Yes	Dermott (1993) theorized that the solar system has a ring just outside of Earth's orbit brought about by resonant trapping of particles. This parameter implements a simple representation of the solar system ring.
volume_emissivity		
	"Filename"	The material file is the database of volume emissivity versus distance from the Sun at 3001 wavelength points.
material_file		
	Formatted	The *Formatted* option allows the database to be an ASCII file, normally *Unformatted* should be chosen for speed.
	Unformatted	The *Unformatted* option requires that the database be converted with the CONVERT program.
volume_scattering		
	"Filename"	A volume scattering function data file in ASCII.
scattering		
	Henyey-Greenstein	Scattering will be performed with a scattering phase function as defined by Hong (1990), a summation of Henyey-Greenstein functions.
	Legendre	Scattering will be performed with a scattering phase function defined by Price and Murdock (1985), a series of Legendre polynomials.

[image]

Parameter	Usage	Description
image		
	No	Suppress image generation (for debugging only).
	Yes	Create an image of the form CBSKYxxx.yyy where xxx is a running numerical index starting at 000. The yyy extension indicates type of image or the header. If yyy is BIN the image is headerless. If the extension is FIT the file is in FITS format. See the related parameters ***output_format*** and ***image_type***.
x_column_pixels		
	Integer	The horizontal image size in pixels. Valid range is 1 to 4096.
y_row_pixels		
	Integer	The vertical image size in pixels. Valid range is 1 to 4096.
pixel_size		
	Floating point	The angular size of each pixel in degrees per pixel. Each pixel is taken to be square.
image_center_longitude_hours		
	Floating point.	The ***image_center_longitude_hours*** parameter specifies the longitude center of the image in decimal hours. For the equatorial system, ***image_center_longitude*** is Right Ascension. The range of values is from 0.0^h to 24.0^h.
	Integer (Hours), Floating Point (Minutes).	Optionally, the ***image_center_longitude_hours*** parameter can specify the longitude center of the image in integer hours and decimal minutes. The range of values is from $0^h\ 0.0^m$ to $24^h\ 0.0^m$.
	Integer (Hours), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_longitude_hours*** parameter can specify the longitude center of the image in integer hours, integer minutes, and decimal seconds. The range of values is from $0^h\ 0^m\ 0.0^s$ to $24^h\ 0^m\ 0.0^s$.

[image] cont.

Parameter	Usage	Definition
image_center_longitude_degrees		
	Floating point.	The ***image_center_longitude_degrees*** parameter specifies the longitude center of the image in decimal degrees. For the equatorial system, ***image_center_longitude*** is Right Ascension that is formally measured in hours. Degrees are obtained by multiplying the Right Ascension, in hours, by 15. The range of values is from 0.0° to 360.0°.
	Integer (Degrees), Floating Point (Minutes).	Optionally, the ***image_center_longitude_degrees*** parameter can specify the longitude center of the image in integer degrees and decimal minutes. The range of values is from 0° 0.0' to 360° 0.0'.
	Integer (Degrees), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_longitude_degrees*** parameter can specify the longitude center of the image in integer degrees, integer minutes, and decimal seconds. The range of values is from 0° 0' 0.0" to 360° 0' 0.0".
image_center_latitude		
	Floating point.	The ***image_center_latitude*** parameter specifies the latitude center of the image in decimal degrees. For the equatorial system, ***image_center_latitude*** is Declination. The range of values is from -90.0° to +90.0°.
	Integer (Degrees), Floating Point (Minutes).	Optionally, the ***image_center_latitude*** parameter can specify the latitude center of the image in integer degrees and decimal minutes. The range of values is from -90° 0.0' to +90° 0.0'.
	Integer (Degrees), Integer (Minutes) Floating Point (Seconds).	Optionally, the ***image_center_latitude*** parameter can specify the longitude center of the image in integer degrees, integer minutes, and decimal seconds. The range of values is from -90° 0' 0.0" to +90° 0' 0.0".
output_format		
	None	The *None* option produces an unformatted image, either byte or word as specified by ***image_type***, with no header.
	FITS	The *FITS* format is standard FITS specified by NASA GSFC FITS Committee. A copy of "Implementation of the Flexible Image Transport System (FITS)", NOST Report 100-03b is included with this documentation.

[image] cont.

Parameter	Usage	Definition
units		
	Jy	Flux units of Jansky
	Jy/sr	Flux units of Jansky/sr
	MJy	Flux units of MegaJansky
	MJy/sr	Flux units of MegaJansky/sr
	W/cm2/sr/um	Flux unit of $W/cm^2/sr/\mu m$.
	W/cm2/sr	Flux unit of $W/cm^2/sr$.
	W/cm2	Flux unit of W/cm^2.
	W/cm2/um	Flux unit of $W/cm^2/\mu m$.
image_type		
	Byte	The output image is stored as an 8-bit, unformatted image.
	4-byte-real	The output image is stored as a 4-byte, unformatted, floating point number. Floating point data are stored in the native format of the computer used to generate images. When transferring the image to a different computer the user must be aware of the need to perform byte swapping.
image_projection		
	Rectangular	The *Rectangular* projection is the familiar Mercator type of projection with parallel lines of latitude and parallel lines of longitude. Maps produced by CBSKY using the *Rectangular* projection do not "wrap" at the poles. This is not the case at 0^h and 24^h were "wrapping" is allowed.
	Gnomonic	The *Gnomonic* projection is the projection of a sphere onto a plane from a tangent point; in other words, as a camera would see it.
	Mollweide	The *Mollweide* projection is an equal-area sinusoidal projection with the lines of latitude all parallel. The *Mollweide* projection is only used as an all sky projection. It is recommended that the ratio of ***x_column_pixels*** to ***y_row_pixels*** be 2:1 for a more pleasing image.

[positional]

The **[positional]** section specifies geometric parameters.

Parameter	Usage	Definition
observer_altitude		
	Floating point	The altitude, in meters, of an observer as defined by the Earth's reference geoid of 1976.
observer_geographic_latitude		
	Floating point	The observer coordinates in degrees referenced to the equator with positive north of the equator and negative south of the equator.
observer_geographic_longitude		
	Floating point	The observer coordinates in degrees relative to Greenwich with positive longitudes taken to be west of Greenwich and negative longitudes east of Greenwich. The possible range is -180° to +180°.
reference_frame		
	J2000	Image coordinates are specified as J2000.
	B1950	Image coordinates are specified as B1950.
	True	Image coordinates are specified as reference to the mean equinox and epoch of the observation.
coordinate_system		
	Equatorial	The equatorial system is the familiar Right Ascension and Declination coordinate system. The sides of the output image are always parallel to the lines of longitude and latitude, that is, no image rotations are possible.
	Galactic	A coordinate system based on the galactic coordinate l_{II} and b_{II} of B1950.
	Ecliptic	The ecliptic coordinate system.
positions		
	Apparent	Apparent positions are those with the ***reference_frame*** given as *True*, the mean equinox and epoch of the observation.
	Astrometric	Astrometric positions are referenced to a standard such as J2000, B1950, B1900, etc.
	Geometric	Geometric coordinates are positions of objects without light time corrections or aberrations.

[positional] cont.

Parameter	Usage	Definition
reference_system		
	Geocentric	A reference system centered on the Earth's center.
	Topocentric	A reference system centered on the position of the observer.
	Heliocentric	A reference system centered on the center of the Sun.
	Barycentric	A reference system centered on the solar system's barycenter.

[spectral]

The **[spectral]** section has four parameters that specify the spectral characteristics of the image. CBZODY6 has a wavelength coverage of 0.2 to 30μm.

Parameter	Usage	Definition
start_wavelength		
	"Filename"	The full path to a filter response file. The data are two columns, wavelength in microns and response. The data do not need to be evenly spaced. A cubic spline is used to interpolate the spectral filter. To aid in the usage of the cubic spline, the first and last values in the response file should be zero (0.0). The ***end_wavelength*** parameter is not used.
	Floating point	The start wavelength, in micrometers, of the bandpass.
	B	Use the standard B photometric band (0.44μm). The ***end_wavelength*** parameter is not used.
	V	Use the standard V photometric band (0.55μm). The ***end_wavelength*** parameter is not used.
	J	Use the standard J photometric band (1.25μm). The ***end_wavelength*** parameter is not used.
	H	Use the standard H photometric band (1.65μm). The ***end_wavelength*** parameter is not used.
	K	Use the standard K photometric band (2.22μm). The ***end_wavelength*** parameter is not used.
	2.4um	Use the COBE/DIRBE 2.4μm band (Band 2). The ***end_wavelength*** parameter is not used.
	12um	Use the IRAS 12μm band (Band 1). The ***end_wavelength*** parameter is not used.
	25um	Use the IRAS 25μm band (Band 2). The ***end_wavelength*** parameter is not used.
	1565A	Use the TD1 1565Å band. The ***end_wavelength*** parameter is not used.
	1400A	Use the TD1 1400Å band. The ***end_wavelength*** parameter is not used.
	1660A	Use the TD1 1660Å band. The ***end_wavelength*** parameter is not used.
end_wavelength		
	Floating point	The stop wavelength, in micrometers, of the bandpass.
delta_wavelength		
	Floating point	The ***delta_wavelength*** parameter specifies the step in wavelength, in micrometers, for an output spectrum.

[spectral] cont.

Parameter	Usage	Definition
flux_mag_zero		
	Floating point	An optional parameter that allows the in-band magnitude of objects to be quoted.
band_center		
	Floating point	If not specified, the CBSD code will define the band center by a simple center of gravity calculation. This is not always in agreement with system developer's definitions. This parameter allows an override of the internal routines.
band_width		
	Floating point	If not specified, the CBSD code will define the bandwidth by a simple center of gravity calculation. This is not always in agreement with system developer's definitions. This parameter allows an override of the internal routines.

[time]

The **[time]** section specifies the date and time of the observation.

Parameter	Usage	Definition
observation_date		
	Integer (Day), Integer (Month), Integer (Year).	The ***observation_date*** is three numbers in the order integer day, numbered month, and 4-digit year. For example, 3 15 1995
	Floating Point	A second option exists for ***observation_date*** which can read the Julian date. Enter the complete Julian date, for example, 2445345.125 In this case the ***observation_time*** is not read.
observation_time		
	Integer (Hour), Integer (Minute), Floating Point (Seconds).	The ***observation_time*** parameter is the integer hour, integer minute, and floating point second of the observation. For example, 12 23 55.06 Ignored if the ***observation_date*** is specified as a Julian date.
time_stamp		
	UT1	The time of observation is defined as the Mean Solar Time.
	UTC	The time of observation is defined as the Coordinated Universal Time - clock time referenced to Greenwich.
	TDT	The time of observation is defined as the Terrestrial Dynamical Time - Used for calculations of ephemerides.

5 Command Line Usage

The CBSD4 codes introduce a command line usage. All parameters in either the CBSKY4 or the CBZODY6 code may be specified on the command line. However, the command line is limited to 512 bytes of total character length so not all arguments can be specified on the command line.

The command line arguments are override arguments. Anything specified in the .inp file is overridden by command line usage. This makes the command line usage useful for batch processing. Specify the problem in the inp file and change location, spectral region, etc. on the command line. An example follows.

cbzody6 start_wavelength="\cbsd4\cbsd\rsr\MSX_A.txt" bandwidth=3.362 band_center=8.276 listfile="F:\MSX_DBF\CB01010000301\Ecliptic_Band_A.strip" log_output=CB01010000301_A

Notice the double quotes (") around path names. These are used for path names or any item with embedded spaces, e.g, "Sun Centered Ecliptic".

6 Troubleshooting

Most problems with the CBSD codes stem from improperly configured input files. Things to check:

1	Is the architecture parameter set properly for your system?
2	Are the paths properly specified? Use \ for DOS/Windows and / for UNIX.
3	Do all paths exist? CBSD4 does not create any directories and all directories must exist prior to execution.
4	Are the datafiles in the data directories specified in the input file? Check for zero length datafiles, an indication of path problems.
5	Did the FITS library compile properly? Check the Makefile for the correct selection of fits modules. The FITS documentation has a test procedure to follow. For Linux systems, the –u parameter in the Makefile should *not* be specified.
6	Path and file names are case sensitive.
7	Files on CDs are stored as read only. You may need to remove the read-only attribute (under DOS/Windows) before making changes to files after they have been transferred to the test platform drive.

References

Cohen, M. (1993). "A Model of the 2-35 μm Point Source Infrared Sky," *Astron. J.*, **105**, 1860.

Cohen, M. (1994a). "Powerful Model for the Point Source Sky: Far-Ultraviolet and Enhanced Midrange Performance," *Astron. J.*, **107**, 1993.

Cohen, M. (1994b). "A Powerful New Model for the Point Source Sky," *Astron. & Space Sci.*, **217**, 181.

Dermott, S. F., P. D. Nicholson, J. A. Burns, and J. R. Houck. (1984). "Origin of the Solar System Dust Bands Discovered by IRAS," *Nature,* **312**, 505.

Dermott, S.F., P. D. Nicholson, J. A. Burns and J. R. Houck. (1985). "Properties and Interaction of Interplanetary Dust," in IAU Colloquium #85, (eds. R. H. Giese and P. Lamy) (D. Reidel Pub. Co.) p395.

Dermott, S.F., P. D. Nicholson, and B. Wolven. (1986). "Preliminary Analysis of the IRAS Solar System Dust Data," in *Asteroids, Comets and Meteors, II* (eds. C-I. Lagerkvist and H. Rickman), Uppsala, p583.

Sykes, M. V. (1988). "IRAS Observations of Extended Zodiacal Structures," *Astrophys. J.,* **334**, L55.

Sykes, M. V. (1990). "Zodiacal Dust Bands - Their Relation to Asteroid Families," *Icarus*, **85**, 267.

Sykes, M. V., Greenberg, R., Dermott, S. F., Nicholson, P. D., Burns, J. A., and Gautier, T. N. (1989). "Dust Bands in the Asteroid Belt," in *Asteroids II*, eds. R. P. Binzel, T. Gehrels, and M. S. Matthews, (Tucson: U. of Arizona), p.336.

Wainscoat, R.J, M. Cohen, K. Vold, H. J. Walker, and D. E. Schwartz (1992). "A Model of the 8-25 Micron Point Source Infrared Sky," *Astrophys. J. Supp.*, **83**, 111.